

Fault Tolerant Control - A Residual based Set-up

Henrik Niemann

Dept. of Elec. Engineering
Automation and Control

Tech. Univ. of Denmark, Build. 326
DK-2800 Kgs. Lyngby, Denmark

hnn@elektro.dtu.dk

Niels Kjølstad Poulsen

Informatics and Math. Modelling
Tech. Univ. of Denmark, Build. 321

DK-2800 Kgs. Lyngby, Denmark

nkp@imm.dtu.dk

Abstract—A new set-up for fault tolerant control (FTC) for stable systems is presented in this paper. The new set-up is based on a simple implementation of the Youla-Jabr-Bongiorno-Kucera (YJBK) parameterization. This implementation of the YJBK parameterization will allow a direct and simple re-configuration of the feedback controller. Another central part of fault tolerant control is fault diagnosis. The controller implementation can be applied directly in connection with both passive diagnosis (PFD) as well as with active fault diagnosis (AFD). The presented FTC set-up is investigated with respect to sensor reconfiguration. Actuator reconfiguration can be dealt with in a similar way.

I. INTRODUCTION

Fault tolerant control (FTC) has become an important area in the last years due to the increasing system complexity. It is important to be able to handle faults in controlled systems - and in a systematic way such that major accidents can be avoided. One of the drawbacks in feedback control is that the effects from a fault, somewhere in the closed loop system, might not be removed from the loop. Furthermore, in many cases, the effect will be amplified through the loop with a major reduction of the performance of the system as the results. In other cases, the closed loop systems will be unstable when faults occur in the loop.

In general, a fault tolerant controller is made up of mainly two parts, a diagnosis part and a controller reconfiguration part. A number of different concepts for FTC has been considered, see e.g. [4], [14], [21].

The focus in this paper is on the application of the YJBK parameterization in connection with FTC. An FTC architecture based on the original implementation of the YJBK parameterization was presented in [14]. One of the advantages in this architecture is that the controller is reconfigured through the YJBK transfer function. In connection with the analysis and design of FTC, this will allow us to use the results from the YJBK and the dual YJBK parameterization. Another important issue with the presented FTC architecture in [14] is the nominal feedback controller is not changed in connection with a reconfiguration. The controller is modified by an additional feedback loop designed with respect to the detected and diagnosed fault.

One of the drawbacks by using the standard implementation of the YJBK parameterization ([1], [2], [5], [19], [20]) as the basis for an FTC architecture is the complexity in the controller switching. A switch from a controller of the same order as the system will in general require a YJBK transfer function of three times the order of the nominal controller, [16]. A new implementation of the YJBK parameterization described in [12] will be the basic for a new FTC architecture. The new implementation of the YJBK parameterization has a more simple structure than the original/standard implementation. This will allow a more simple way to switch between different controllers through the YJBK transfer function. This new controller implementation exists in two versions as in the original implementation, [5]. Both versions of the new implementations have a structure that includes a residual vector. These residual vectors are used for an internal feedback in the controllers. Further, these residual vectors are also used directly in connection with the fault diagnosis in the FTC architecture. In the original implementation of the YJBK parameterization, only one of the versions include a residual vector can be applied in connection with fault diagnosis. It will therefore only be possible to get one FTC architecture based on the original implementation of the YJBK parameterization.

In this paper, the new FTC architecture will be based on the dual version of the implementation of the YJBK parameterization. It will be shown how it is possible to obtain both passive and active fault diagnosis and also controller reconfiguration based on the same set of input and output vectors.

Another relevant aspect in relation with FTC is the possibility to change sensors and actuators in connection with controller reconfiguration. When a fault has occurred in the system, it will in many cases be relevant to let the reconfigured controller use another set of actuators and sensors than the nominal controller. It is then possible to use more reliable actuators and sensors in the faulty case than in the fault free case. The case of changing sensors in connection with controller reconfiguration has been considered in the last part of the paper. The derived results can be extended to the general case where also actuator changes.

The rest of this paper is organized as follows. Some preliminary results are given in Section II. Implementation of controllers are considered in Section III followed by fault diagnosis in Section IV. The FTC implementation is described in Section V. Change of sensors in connection with controller reconfiguration is considered in Section VI. The paper is closed by a Conclusion in Section VII.

II. PRELIMINARY RESULTS

A. System set-up

The system set-up is shortly introduced in the following. Let a general continuous-time stable MIMO system be given by:

$$\Sigma_P : \begin{cases} e = G_{ed}(\theta)d + G_{eu}(\theta)u \\ y = G_{yd}(\theta)d + G_{yu}(\theta)u \end{cases} \quad (1)$$

where $d \in \mathcal{R}^r$ is a disturbance signal vector, $u \in \mathcal{R}^m$ the control input signal vector, $e \in \mathcal{R}^q$ is the external output signal vector to be controlled and $y \in \mathcal{R}^p$ is the measurement vector. The system description in (1) may depend on a number (k) of parameters. Let θ_i , $i = 1, \dots, k$ denote the parametric deviations away from the nominal values, i.e. $\theta_i = 0$, $i = 1, \dots, k$ in the nominal case. Let us for short arrange the deviations in a vector:

$$\theta = (\theta_1, \dots, \theta_i, \dots, \theta_k)^T$$

Furthermore, let

$$\vartheta_i = (0, \dots, \theta_i, \dots, 0)^T$$

which represent the situation with a fault or change in precisely one parameter. In many cases it will be possible to give an explicit expression of the connection between the system and the parametric faults (see [9], [10]).

Let the system be controlled by a stabilizing feedback controller given by:

$$\Sigma_C : \{ u = Ky \} \quad (2)$$

B. Coprime factorization

Let a coprime factorization of the system G_{yu} from (1) and the stabilizing controller K from (2) be given by:

$$\begin{aligned} G_{yu} &= N_0 M_0^{-1} = \tilde{M}_0^{-1} \tilde{N}_0, \quad N_0, M_0, \tilde{N}_0, \tilde{M}_0 \in \mathcal{R} \mathcal{H}_\infty \\ K &= UV^{-1} = \tilde{V}^{-1} \tilde{U}, \quad U, V, \tilde{U}, \tilde{V} \in \mathcal{R} \mathcal{H}_\infty \end{aligned} \quad (3)$$

The stability condition gives that:

$$(\tilde{V}M_0 - \tilde{U}N_0)^{-1} = Z \in \mathcal{R} \mathcal{H}_\infty \quad (4)$$

or

$$(\tilde{V}M_0 Z - \tilde{U}N_0 Z) = (\tilde{V}M - \tilde{U}N) = I$$

where the related right coprime factorization of G_{yu} is given by:

$$G_{yu} = (N_0 Z)(M_0 Z)^{-1} = NM^{-1}, \quad N, M, Z \in \mathcal{R} \mathcal{H}_\infty \quad (5)$$

Equivalent for the left coprime factorization of G_{yu} is given by:

$$G_{yu} = (\tilde{Z}\tilde{M}_0)^{-1}(\tilde{Z}\tilde{N}_0) = \tilde{M}^{-1}(\tilde{N}), \quad \tilde{N}, \tilde{M}, \tilde{Z} \in \mathcal{R} \mathcal{H}_\infty \quad (6)$$

where \tilde{N} and \tilde{M} will satisfy:

$$(-\tilde{Z}\tilde{N}_0 U + \tilde{Z}\tilde{M}_0 V) = (-\tilde{N}U + \tilde{M}V) = I \quad (7)$$

The eight matrices N , M , \tilde{N} , \tilde{M} , U , V , \tilde{U} and \tilde{V} from (3), (5) and (6) will satisfy the double Bezout equation given by, see [19]:

$$\begin{aligned} \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} &= \begin{pmatrix} \tilde{V} & -\tilde{U} \\ -\tilde{N} & \tilde{M} \end{pmatrix} \begin{pmatrix} M & U \\ N & V \end{pmatrix} \\ &= \begin{pmatrix} M & U \\ N & V \end{pmatrix} \begin{pmatrix} \tilde{V} & -\tilde{U} \\ -\tilde{N} & \tilde{M} \end{pmatrix} \end{aligned} \quad (8)$$

Further, Z and \tilde{Z} are given by:

$$\begin{aligned} Z &= M_0^{-1}M \in \mathcal{R} \mathcal{H}_\infty \\ \tilde{Z} &= \tilde{M}\tilde{M}_0^{-1} \in \mathcal{R} \mathcal{H}_\infty \end{aligned} \quad (9)$$

C. The YJBK Parameterization

With the previous mentioned coprime factorization of the system G_{yu} and the controller K , we can give a parameterization of all controllers that stabilize the system in terms of a stable transfer function Q , i.e. all stabilizing controllers are given by using a right factored form [19]:

$$K(Q) = (U + MQ)(V + NQ)^{-1}, \quad Q \in \mathcal{R} \mathcal{H}_\infty \quad (10)$$

or by using a left factored form:

$$K(Q) = (\tilde{V} + Q\tilde{N})^{-1}(\tilde{U} + Q\tilde{M}), \quad Q \in \mathcal{R} \mathcal{H}_\infty \quad (11)$$

Using the Bezout equation, the controller given either by (10) or by (11) can be realized as an LFT (linear fractional transformation) in the parameter Q :

$$K(Q) = \mathcal{F}_l \left(\begin{pmatrix} UV^{-1} & \tilde{V}^{-1} \\ V^{-1} & -V^{-1}N \end{pmatrix}, Q \right) = \mathcal{F}_l(J_K, Q) \quad (12)$$

where (12) is the same for both the right and the left form given in (10) and (11), respectively.

In the same way, it is possible to derive a parameterization in terms of a stable transfer function S of all systems that are stabilized by one controller, i.e. the dual YJBK parameterization. Using the right form, the parameterization is given by [19]:

$$G_{yu}(S) = (N + VS)(M + US)^{-1}, \quad S \in \mathcal{R} \mathcal{H}_\infty \quad (13)$$

or by using a left factored form:

$$G_{yu}(S) = (\tilde{M} + S\tilde{U})^{-1}(\tilde{N} + S\tilde{V}), \quad S \in \mathcal{R} \mathcal{H}_\infty \quad (14)$$

An LFT representation of (13) or (14) is given by:

$$\begin{aligned} G_{yu}(S) &= \mathcal{F}_l \left(\left(\begin{array}{cc} NM^{-1} & \tilde{M}^{-1} \\ M^{-1} & -M^{-1}U \end{array} \right), S \right) \\ &= \mathcal{F}_l(J_G, S) \end{aligned} \quad (15)$$

Further, S is given by, [19]:

$$S = \mathcal{F}_u(J_K, G_{yu}(S)) \quad (16)$$

The dual YJBK transfer function S will be a function of the system variations θ , i.e. $S = S(\theta)$. The connection between S and θ has been considered in details in [7]. Assuming that $\theta = 0$ is the nominal value of θ , then there will exist the following simple relation, [7]:

$$S(\theta) = 0, \text{ for } \theta = 0 \quad (17)$$

This relation will be applied in the following in connection with fault diagnosis.

III. CONTROLLER IMPLEMENTATION

The implementation of the YJBK parameterization can be done as described in e.g. [8], [9], [14], [19] and the dual implementation in [5]. These implementations include an inversion of \tilde{V} or V , respectively. Further, a switch from a nominal controller K to another controller K_i through a use of the YJBK transfer function Q result in a quite complex Q .

Some of these drawbacks with the normal implementation can be removed by using a new implementation of the YJBK parameterization. The implementation has been described in details in [12]. The implementation is shown in Fig. 1. A dual version is shown in Fig. 2. Both implementations are based on residual vectors, ε or $\tilde{\varepsilon}$ respectively, as described in next section. The two implementations are therefore called for residual based implementations.

$U(Q)$ and $\tilde{U}(Q)$ are (see (10) and (11)) given by:

$$\begin{aligned} U(Q) &= U + MQ \\ \tilde{U}(Q) &= \tilde{U} + Q\tilde{M} \end{aligned} \quad (18)$$

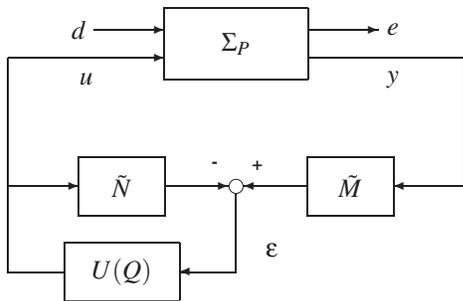


Fig. 1. A residual based implementation of the YJBK parameterization.

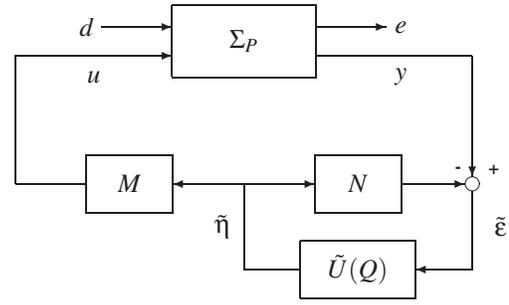


Fig. 2. A dual implementation of the YJBK parameterization shown in Fig. 1.

A. Controller Switching

Based on the YJBK parameterization, it is possible to switch from a nominal controller K to another controller K_i by a suitable selection of Q . Assume the existence of a coprime factorization of the system and the new controller

$$\begin{aligned} G_{yu} &= (NZ_i)(MZ_i)^{-1} = N_iM_i^{-1}, \quad Z_i = M^{-1}M_i \\ &= (\tilde{Z}_i\tilde{M})^{-1}(\tilde{Z}_i\tilde{N}) = \tilde{M}_i^{-1}\tilde{N}_i, \quad \tilde{Z}_i = \tilde{M}_i\tilde{M}^{-1} \\ K_i &= U_iV_i^{-1} = \tilde{V}_i^{-1}\tilde{U}_i \end{aligned}$$

which satisfy the double Bezout equation given by:

$$\begin{aligned} \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} &= \begin{pmatrix} \tilde{V}_i & -\tilde{U}_i \\ -\tilde{N}_i & \tilde{M}_i \end{pmatrix} \begin{pmatrix} M_i & U_i \\ N_i & V_i \end{pmatrix} \\ &= \begin{pmatrix} M_i & U_i \\ N_i & V_i \end{pmatrix} \begin{pmatrix} \tilde{V}_i & -\tilde{U}_i \\ -\tilde{N}_i & \tilde{M}_i \end{pmatrix} \end{aligned} \quad (19)$$

Then a switching from K to K_i can be obtained by using Q_i given by ([16]):

$$Q_i = Z_i(\tilde{U}_iV - \tilde{V}_iU) \text{ or } Q_i = (\tilde{V}U_i - \tilde{U}V_i)\tilde{Z}_i \quad (20)$$

in (10) or in (11).

If the residual based implementation in Fig. 1 and Fig. 2 is applied, then it is impossible to reduce the complexity of the implementation of Q_i significantly. Let $U(Q)$ in (18) be given by:

$$U(Q_i) = U + \alpha MQ_i$$

where K is obtained for $\alpha = 0$ and K_i for $\alpha = 1$. Including Q_i in $U(Q)$ gives:

$$U(Q_i(\alpha)) = (1 - \alpha)U + \alpha U_i\tilde{Z}_i \quad (21)$$

Equivalent, using $\tilde{U}(Q)$ in Fig. 2 gives

$$\tilde{U}(Q_i(\alpha)) = (1 - \alpha)\tilde{U} + \alpha\tilde{Z}_i\tilde{U}_i \quad (22)$$

The equivalent Q_i is given by

$$Q_i = M^{-1}(U_i\tilde{Z}_i - U) \text{ or } Q_i = (Z_i\tilde{U}_i - \tilde{U})\tilde{M}^{-1} \quad (23)$$

The closed loop transfer function from external input d to external output e is given by:

$$\begin{aligned} T_{ed} &= G_{ed} + G_{eu}U\tilde{M}G_{yd} + G_{eu}MQ\tilde{M}G_{yd} \\ &= G_{ed} + G_{eu}U(Q)\tilde{M}G_{yd} \end{aligned} \quad (24)$$

when the general feedback controller $K(Q)$ is applied.

Including Q_i given by (23) in (24) gives directly:

$$T_{ed} = G_{ed} + G_{eu}U_i\tilde{M}_iG_{yd} \quad (25)$$

which show that the controller has been switched from K to K_i .

IV. FAULT DIAGNOSIS

In connection with fault diagnosis it is possible to apply both implementations of the feedback controller shown in Fig. 1 and in Fig. 2. In the following, the fault diagnosis will be considered only in connection with the controller implementation in Fig. 2.

Let the YJBK transfer function Q be removed from the set-up in Fig. 2. Further, including a reference r input gives the set-up shown in Fig. 3. η is an auxiliary input applied in connection with active fault diagnosis.

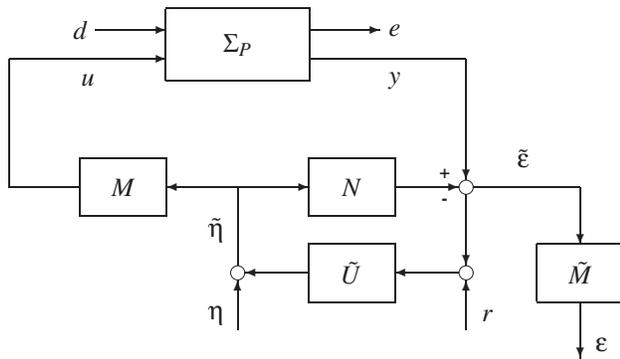


Fig. 3. A set-up for both passive and active fault diagnosis in closed-loop.

The transfer function from the inputs, d , r and η to the residual vector ε can be derived in the following way:

$$\begin{aligned} \varepsilon &= \tilde{M}(I + N\tilde{U})^{-1}(y - N\tilde{U}r - N\eta) \\ &= V^{-1}(y - N\tilde{U}r - N\eta) \end{aligned}$$

Further, the output y is given by:

$$\begin{aligned} y &= G_{yd}(\theta)d + G_{yu}(\theta)M(\eta + \tilde{U}r + \tilde{U}\tilde{M}^{-1}\varepsilon) \\ &= G_{yd}(\theta)d + G_{yu}(\theta)(M\eta + M\tilde{U}r + U\varepsilon) \end{aligned}$$

This result in the following equation for ε :

$$\varepsilon = (\tilde{M} + S\tilde{U})G_{yd}(\theta)d + S\tilde{U}r + S\eta \quad (26)$$

where (13) and (14) has been applied in calculation of (26).

The residual vector ε can be applied in connection with both passive and active fault diagnosis. In the passive approach ($\eta = 0$) the residual vector is given by:

$$\varepsilon = \tilde{M}G_{yd}(0)d \quad (27)$$

in the nominal case ($\theta = 0$). Changes in the system can be detected by using statistical tests directly on the residual vector to detect changes in the statistical properties such as mean and variance. Tests such as CUSUM and maximum likelihood methods can be applied, see e.g. [3], [6].

In the active approach, the auxiliary input η is applied in order to enhance the precision of the diagnosis, which is based directly on the condition on S given by (17). In the case of a fault, S will be non-zero, and the detection is carried out a detection of the signature of η in the residual. Fault isolation can be developed by an investigation of change of gain and phase properties through S . These changes depend directly of the parametric faults in the system. S plays a central role in the AFD approach and has therefore also been named *the fault signature matrix*, [9], [11], [17], [18], wherein the AFD approach is described in details.

Note that the implementation in Fig. 3 based on the block diagram in Fig. 2 gives directly a residual vector that can be applied in connection with both passive and active fault diagnosis. This is not obtained in the original dual implementation of the YJBK parameterization. It has been pointed out in [21] that the missing naturally residual vector in the set-up will make this implementation useless in connection with fault detection and fault tolerant control. This problem has been handled by using the new implementation in Fig. 2 or Fig. 3.

V. FTC IMPLEMENTATION

The implementation of a fault tolerant controller based on the controller implementation in Section III and the fault diagnosis part in Section IV is now described. The FTC implementation will be based on the dual implementation as shown in Fig. 2, because both controller switching and fault diagnosis can be derived from this architecture. The structure of a fault tolerant controller based on this controller implementation is shown in Fig. 4.

It is clear from the FTC set-up in Fig. 4 that both the reconfiguration and the diagnosis part are based on the same set of input and output vectors, i.e. (η, ε) . The set-up is shown in Fig. 4 is based on an AFD method. A passive diagnosis method can be included without changing the structure.

An important aspect of the shown set-up is that the structure of the diagnosis part is independent of the applied feedback controller. A reconfiguration of the feedback controller in terms of a non-zero Q will not change the structure of the diagnosis part.

The design of the reconfiguration part Q is not restricted to a specific method. A number of Q 's can be designed off-line or it is possible to using on-line optimization methods. It will depend on the specific application. Further, an advantage with the FTC set-up is that the nominal controller is not

